REMARKS

Claims 1, 3-8, 11-15, 17-19, 22-32, 34-36, 38 and 39 are pending in the present application. Applicants respectfully request reconsideration of the application in view of the above amendments and remarks made herein.

I. Rejections Under 35 U.S.C. § 103

Claims 1, 3-8, 11-15, 17-19, 22-32, 34-36, 38 and 39 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 5,797,007 issued to *Erickson et al.* (hereinafter "*Erickson*") in view of U.S. Patent Application Publication No. 2003/0140308 attributed to *Murthy et al.* (hereinafter "*Murthy*"), in further view of U.S. Patent No. 6,470,344 issued to *Kothuri et al.* (hereinafter "*Kothuri*"), for the reasons set forth on pages 2-9 of the Office Action.

The independent claims are Claims 1, 15, 26 and 38.

Claims 1, 15 and 38 claim, *inter alia*, "generating automatically an interface for the persistent storage structure, wherein the interface comprises access object classes that are generated automatically to enable management of object instance data in the persistent storage structure, wherein each object is associated with a respective access object class for performing methods including a method for deleting the object instance data associated with the respective access object class." Claim 26 claims, *inter alia*, "an access interface comprising access object classes that are generated automatically to enable management of object instance data in the persistent storage structure, wherein each object is associated with a respective access object class for performing methods including a method for deleting the object instance data associated with the respective access object class."

Erickson teaches a single base class, SOMObject, which includes general methods that are inherited by all of its subclasses (see column 3 lines 66-67, column 4 lines 1-10). *Erickson* teaches a DefaultEncoderDecoder class, which includes a "get" and "set" method for use by all objects, to implement flexible, generalized storage of object information (see column 4 lines 55-67). Implementation of general methods for use with all objects in a persistent storage medium, as taught by *Erickson*, is clearly not analogous to an object associated with a respective access

object class for performing methods with the respective access object class, as essentially claimed in Claims 1, 15, 26 and 38. Therefore, *Erickson* fails to teach or suggest all of the limitations of Claims 1, 15, 26 and 38.

Murthy teaches a method allowing users to register XML schemas in a database system and mapping constructs defined in the XML schema to constructs supported by the database system. *Murthy* does not teach methods for use with objects in a persistent storage structure, let alone an object associated with a respective access object class for performing methods with the respective access object class, as essentially claimed in Claims 1, 15, 26 and 38. Therefore, *Murthy* fails to cure the deficiencies of *Erickson*.

Kothuri teaches update operations, such as the insertion or deletion of data, to carry out the removal of a data item from a database and to update a corresponding index (see column 16 lines 63-66). These general update operations, which are available for use with all data in the database, are clearly not analogous an object associated with a respective access object class for performing methods with the respective access object class, as essentially claimed in Claims 1, 15, 26 and 38. Consider that the general update operation of *Kothuri* is not associated with a respective object in a persistent storage system, essentially as claimed in Claims 1, 15, 26, and 38. Therefore, *Kothuri* fails to cure the deficiencies of *Erickson* and *Murthy*.

The combination of *Erickson, Murthy* and *Kothuri* teaches implementing general methods in a base class for use with subclasses, methods allowing users to register XML schemas in a database system, and general update operations for the insertion or removal of data from a database and an index. The combination clearly does not teach or suggest associating each object in a persistent storage structure with an access object class for performing methods specifically designed for use with a respective object, as essentially claimed in Claims 1, 15, 26 and 38. Accordingly, the combination does not teach each and every limitation of Claims 1, 15, 26 and 38.

Further still, the combination of *Erickson*, *Murthy*, and *Kothuri* does not teach the automatic creation of a method enabling an object to delete its own object instance data, as essentially claimed in Claims 1, 15, 26 and 38. Particularly, *Erickson* teaches encoding and decoding object instance data by implementing "get" and "set" methods to control the storage and restoration of persistent data. The Examiner points to column 4 lines 55-65 of *Erickson* as

teaching automatically creating methods for one of storing, retrieving, searching and removing object instance data in a persistent storage medium. Respectfully, *Erickson* merely teaches using a "get" method for controlling storage of an object and using a "set" method for controlling restoration of an object (see column 4 lines 63-67, column 5 lines 1-5). The methods for retrieval and restoration of persistent objects, as taught by *Erickson* (see column 2 lines 40-50), is clearly not analogous to the automatic creation of a method that enables an object to delete its own object instance data, as essentially claimed in Claims 1, 15, 26 and 38. Therefore, *Erickson* fails to teach or suggest all of the limitations of Claims 1, 15, 26 and 38.

Murthy teaches methods allowing a user to register XML schemas in a database and delete XML schemas from a database (see paragraph [0081]). *Murthy* does not teach the automatic creation of a method enabling an object to delete its own object instance data, as essentially claimed in Claims 1, 15, 26 and 38. Allowing a user to invoke a method to delete an XML schema is clearly not analogous to the automatic creation of a method that enables an object to delete its own object instance data without input from the user, as essentially claimed in Claims 1, 15, 26 and 38. Furthermore, *Murthy* teaches away from the automatic creation of a method enabling an object to delete its own object instance data, as the method taught by *Murthy* includes a failsafe preventing the user from deleting a schema if other objects in the database are dependent upon it (see paragraph [0081]). Therefore, *Murthy* fails to teach or suggest all of the limitations of Claims 1, 15, 26 and 38.

Kothuri teaches methods for buffering nodes of a hierarchical index during operations on multi-dimensional data represented by the index, as well as methods that are aimed at performing query operations. *Kothuri* does not teach the automatic creation of a method enabling an object to delete its own object instance data, as essentially claimed in Claims 1, 15, 26 and 38. *Kothuri* teaches comparing two buffered nodes to determine which node provides less caching benefit and removing that node from a buffer (see column 4 lines 55-67, column 5 lines 1-15). Removing a node with the least caching benefit from a buffer is clearly not analogous to the automatic creation of a method that enables an object to delete its own object instance, as essentially claimed in Claims 1, 15, 26 and 38. Therefore, *Kothuri* fails to cure the deficiencies of *Erickson* and *Murthy*.

The combination of *Erickson, Murthy* and *Kothuri* teaches "get" and "set" methods for controlling storage and restoration of an object, a method allowing a user to delete an XML schema from a database, and a method that removes a node with the least caching benefit from a buffer. The combination clearly does not teach or suggest the automatic creation of a method enabling an object to delete its own object instance data, as essentially claimed in Claims 1, 15, 26 and 38. Accordingly, the combination does not teach each and every limitation of Claims 1, 15, 26 and 38.

Therefore, for at least the reasons above, Claims 1, 15, 26 and 38 are believed to be patentable and non-obvious over the combination of *Erickson*, *Murthy* and *Kothuri*. Applicants respectfully submit that inasmuch as Claims 3-8, 11-14, 17-19, 22-25, 27-32, 34-36 and 39 are dependent on Claims 1, 15, 26 and 38, and Claims 1, 15, 26 and 38 are patentable over the cited references, Claims 3-8, 11-14, 17-19, 22-25, 27-32, 34-36 and 39 are patentable as dependent on patentable independent claims. Withdrawal of the instant rejections is respectfully requested.

CONCLUSION

In view of the foregoing, it is believed that all claims now pending patentability define the subject invention over the prior art of record and are in condition for allowance. Early and favorable reconsideration of the case is respectfully requested.

Respectfully submitted,

Date: June 19, 2008

By: /Nathaniel T. Wallace/
/Nathaniel T. Wallace/
Nathaniel T. Wallace
Reg. No. 48,909
Attorney for Applicant(s)

F. Chau & Associates, LLC
130 Woodbury Road
Woodbury, New York 11797
TEL: (516) 692-8888
FAX: (516) 692-8889